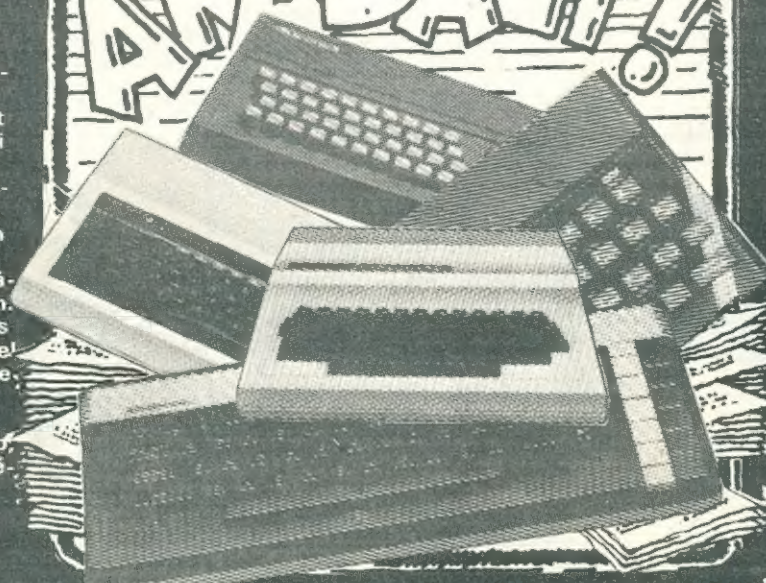


GÉP

AKADÁLY!



Nemrég egy barátommal a kisvállalkozások számítógépesítéséről beszélgettünk.

A barátom maga is kisvállalkozó, mégpedig a kisvállalkozások kisvállalkozója, azaz abból áll, hogy más vállalkozások könyvelési, gazdálkodási ügyeit intézi, s ezekben ad hasznos és haszontalan tanácsokat. Gondoltam, hogy számszaki embert nem lehet nehéz meggyőzni a személyi számítógép hasznosságáról, hát úgy fordítottam a beszélgetés menetét, hogy szóba hoztam, vajon ők miért nem használnak számítógépet.

Nem lenne-e érdemes munkájukat gépesíteni – kérdeztem. Egyfelől magát a mechanikus számolási műveletsort géppel végeztetni, másrészt a könyvelési „trükköket”, lehetőségeket gépre vinni, s ily módon már egy bonyolultabb, nagyobb döntést igénylő tevékenységben tanácsadóként igénybe venni a gépet mint segítőtársat.

Hiszen logikusnak tűnik a gondolat, hogy egy adóbevallás elkészítésében nem egyféle variáció létezik, vannak dolgok, amelyek így is könyvelhetők, meg úgy is, s a szabályok, a rendeletek is adnak lehetőséget bizonyos döntések legelőnyösebb meghozatalára. Ezenkívül kintről érkezett információink is vannak arról, hogy efféle munkákban sokfelé használnak már számítógépet, s nem is egyféle kész programot árulnak ilyen célra.

Lévn, hogy nem vagyok gazdasági szakember, nem tudhattam, s nem is gondoltam, hogy eme logikusnak tűnő gondolatom megvalósulása miféle akadályba ütközik. Nem tudtam, s így azt sem értettem, hogy miért nem árulnak számítástechnikai kisvállalkozások tucatnyiféle könyvelőprogramot, adóbevallási tanácsadót, döntéselőkészítőt nálunk is. A válasz lényegesen egyszerűbb, mint gondoltam. Egyszerűen nem tartja érdemesnek sem a programozó, sem a könyvelő gm, hogy ilyen programot kidolgozzon, kidolgoztasson. Mert mint barátom mondta, mire egy ilyen program elkészülne, mire a használhatósága beigazolódna, s ily módon piaca lenne, a programot már el lehetne dobni, s helyette újat írni. A könyvelésben, az adózásban követendő szabályok ugyanis oly

gyakran változnak, hogy gyakorlatilag évente kellene új programot írni.

Valóban, gyakran olvasunk szaklapokban – így az Ötletben is – a szabályozók túl gyakori változásáról, a jogszabályok irdatlan dzsungeléről.

S mindez persze már nemcsak a kisvállalkozások, s nemcsak a könyvelés gépesítését nehezíti. A számítógép

hasznos eszköz lehet, lehetne a gazdaság legkülönbözőbb szféráiban, az ügyvitelben éppúgy mint a műszaki munkában vagy másutt.

Am ahhoz, hogy ez a hasznos eszköz hasznot hajtson, hogy termelőerővé váljon, ahhoz programok kellene.

A programhoz pedig sokféle szabványra, szabályozóra, rendeletre, előírásra oda kell figyelni.

Márpedig – s ez is hozzátartozik az igazsághoz – programot, komoly elemző, döntéselőkészítő vagy éppen elemző munkát végző, segítő programot nem lehet két perc alatt s két forintért elkészíteni.

Épp így nem lehet egy jól elkészített programot percenként átírni, alakíttatni.

Ma ott tartunk, hogy a gépek kezdenek megszokottá, mindennapos eszközzé válni nálunk is. Ma ott tartunk, hogy keressük azokat a területeket, konkrét feladatokat, amelyekben a számítógép visszatérítheti az árát. S ma ott tartunk, hogy ennek a tevékenységnek a kiteljesedéséhez bizonyos feltételek megteremtését alapvető célként kell elfogadni. S ezek közé a feltételek közé eszerint föl kell vennünk azt is, hogy bizonyos szabályozók ritkábban kellene hogy változzanak. S hogy tovább menjek, ezt el is kellene hitetni az emberekkel. Azaz ahhoz, hogy a barátom meg merjen rendelni nem kis pénzért egy programot egy számítástechnikai kisvállalkozástól, ehhez hinnie kellene abban, hogy az adózással, könyveléssel kapcsolatos jogszabályok a következő három évben nem változnak. De tessék megmondani, hogy a rugalmas gazdaságpolitika változó szabályozóit ismerve, bizhatunk-e ilyesmiben? Hát tessék, ismert egy skizofrén dilemma, amelynek megoldásában tanácsot kér olvasóitól a szerkesztő:

Angyalosi László

BELÜLRŐL

- 18 **Hiroldal** – amelyből megtudhatják, hogy nem minden Commodore 64 – ami úgy néz ki
- 20 **Zsebszámológép a HT-n** – a gépnyerő pályázat első feladata legjobb programjainak egyike, nem minden tanulság nélkül való
- 22 **Programajánlat** – egy játék a Primóra Futkározás címmel
- 23 **Vallató-hozzászólás** – egy olvasónk megírja mindazt, amit a gépkönyvből nem tudott meg, de rájött
- 24 **Sorvezető** – lebegőpontos aritmetika a Spectrumon II. – egy írás, amely már a múlt hónapban elkezdődött
- 25 **Puska a HT-re** – amely egy lapon tartalmazza a HT BASIC legfontosabb tudnivalóit
- 27 **Tavalyi hó** – némi összegzés a tavalyi év mikroszámítógépes történéseiről – a Byte alapján
- 28 **Hardverötletek** – ezúttal egy olyan ötlet, amely a C 64 főhasználóit gyakori bosszúságtól szabadítja meg! Nincs többé lemerevedett Commodore
- 29 31 **Posta** – amelyben elbújtattuk telefonos szolgálatunk legközelebbi időpontját is
- 32 **Primo-nyerő** – a második feladat sem nehezebb, mint az első volt!

HÍROLDAL

Gépi fordítás!

Óránként háromezer szavas sebességgel képes japán nyelvről angolra fordítani egy a közelmúltban Japánban kifejlesztett számítógépes fordító berendezés. A közel százezer dollárt érő technikai újdonság azonban a finomságokat kevésbé képes érzékelni, ezért szükség van még az ember közreműködésére, aki a lefordított szöveget végleges formára átírja, szerkeszti.

Vámtestet büntette!

Egy Miami-ban dolgozó vámtiszt lesz az első áldozata az új számítógépes titokvédelmi törvénynek, amely 1984 októberében lépett hatályba az Egyesült Államokban.

A vámtisztet tavaly októberben tartóztatták le azzal a váddal, hogy illegálisan kapcsolódott egy állami számítógéprendszerre és felhatalmazás nélkül adatokat módosított annak adatbázisában. Mint a nyomozás során kiderült, az adatmódosítás nem öncélú volt, hiszen a vádlottat és két büntársát 160 000 dollárhoz juttatta. Az új törvény szigorúságára jellemző, hogy ezért a büntettért 25 év börtön és 30 000 dollár pénzbírság szabható ki.

Teszteredmény.

Mint ismeretes, az IBM 1984 augusztusában jelentette be az IBM PC AT számítógépét. A géppel együtt belgért hálózati szoftver, az ezt támogató PC-DOS 3.1 operációs rendszer verzió, a XENIX többfelhasználós változata késik, és csak 1985 első negyedévére ígérik ezeknek a termékeknek a piacra hozatalát.

A Personal Computing számítástechnikai szaklap mindazonáltal megvizsgált egy IBM

PC AT mikroszámítógépet 20 Mbyte-os merev lemezzel és 1,2 Mbyte-os floppy lemez háttértárral. A próbák során kiderült, hogy az IBM PC AT, 80286-os mikroprocesszorának köszönhetően, sebességét legalább kétszeresére növelte. Ez elegendőnek bizonyult ahhoz, hogy az IBM riválisa, az AT & T személyi számítógépéhez viszonyítva is gyorsabb legyen.

Basic tanfolyam!

Új BASIC-tanfolyamok indultak januárban és februárban többek között a MOM Művelődési Központban, a kőbányai Pataky István Művelődési Központban és a Gáz-művek óbudai Művelődési Házában. Míg a MOM-ban csak általános iskolásoknak szervezték a programot, addig a másik két tanfolyamon nincs korhatári kötöttség. A kőbányai tanfolyamot nyolchetesre tervezték, melynek a részvételi díja nyolcszáz forint.

Fedélzeti...

Kis fedélzeti számítógépet fejlesztett ki gabonakombájnokokhoz a mezőúri Magyar-Mongol Barátság Tsz szakembercsoportja. A kombájn vezetőfülkéjében elhelyezett berendezés folyamatos információt szolgáltat a vezetőnek a kombájn egyes egységeinek állapotáról, javítva ezáltal az elvégzett munka minőségét.

Szoftver bolt!

A Skála Budapest Nagyruház második emeletén mikroszámítógépes szoftvert árusító bolt kezdte meg működését a közelmúltban. A Commodore 64 személyi számítógépre készült programok segítségével számos gazdálkodási, nyilvántartási feladatot oldhatnak meg vállalataink, intézményeink.

Közlekedés

Tizenkilenc mikroszámítógépet vettek használatba a Budapesti Közlekedési Vállalat központi részlegeiben. A mikrogépek alkalmazásától a főváros közlekedését meghatározó vállalat szolgáltatásainak további javulását várják. Hasznos segítséget nyújthatnak az új gépek a forgalomszervezés, a gépjárműüzemeltetés, az anyaggazdálkodás, a menetrendszámítás, a munkaidőbeosztás területén.

Tehénész szoftver

A Szigetvári Állami Gazdaság tehénészeti takarmányoptimalizálási szoftver, illetve számítógép segítségével biztosítják, hogy a gazdaság görögországi üzemében minden tehén az optimális abrakadagot kapja. Az új módszer bevezetésével jelentősen nőtt az állomány tejtermelése, miközben csökkent a takarmányfelhasználás.

RADELKIS rajzoló

OH-860 típusjelzéssel digitális rajzológép sorozatgyártását kezdték meg a Radelkis Szövetkezetben. A fontos importkiváltó számítástechnikai eszköz jellemzője, hogy minden hazai forgalomban levő számítógéphez csatlakoztatható. Különböző számítástechnika-alkalmazó vállalatok és intézmények megrendelésére mintegy száz darab berendezés készül el a szövetkezetben a második negyedév végéig.

Mini képernyő!

A Mitsubishi Tredia notesz nagyságú számítógép azzal keltett meglepetést, hogy 80 soros, 25 oszlopos folyékony kristályos képernyőt ad 24 kbyte-os RAM-számítógépéhez. A konfigurációhoz tartozik még egy mikrokazettás olvasó, egy modem, egy vonalkód-olvasó és egy ROM cartridge csatlakoztatási lehetőség. A belső elemek lehetővé teszik a gép 8 órás működését. A gép ára 1200 dollár.

- **bit**: egy kettes számrendszerbeli helyiérték (0 vagy 1)
- **byte** (bájt): 8 bitből álló memória „egység”
- **gépi kód**: a gép saját nyelve, a BASIC utasításokat először erre fordítja le, csak azután tudja végrehajtani
- **hardware** (hárduer): a gép műszaki-fizikai „teste”
- **interface** (interféjsz): más gépekhez vagy perifériákhoz való kapcsolódási lehetőség
- **memória**: adatok és programok tárolására szolgáló egység
- **mikroprocesszor** (CHIP): a mikrogép „lelke”, a gép működését vezérlő integrált áramkör
- **nagy felbontású grafika**: ha a gép a képernyőn sok pontot tud külön megjeleníteni
- **periféria**: a géphez csatlakoztatható megjelenítő, tároló és adatbeviteli eszközök
- **program**: feladat végrehajtására összeállított utasítássorozat
- **RAM** (angol betűszó): a gépet használó számára teljesen hozzáférhető (felülírható és kiolvasható) memóriaterület
- **ROM** (angol betűszó): csak kiolvasható memóriaterület, amely a gép programozhatóságát biztosító „tudásanyagot” tartalmazza
- **software** (szoftver): mindaz, ami a gépbe „beleírható”
- **szintaxis**: a programíráshoz vonatkozó formai szabályok összessége



ÚJ!



A Personal Computing Today kőpadra fektette a Commodore új adóját, a C16-ot. A gép műszaki adatai röviden: 6502 kompatibilis 7501 típusú mikroprocesszor, 16 K RAM, melyből 12 K BASIC munkaterület. Kétféle grafikus üzemmód: 320x200 képpont és négy szín minden karaktermezőben, vagy 160x200 képpont és négy szín minden karaktermezőben. Hangkeltési lehetőség: két darab négyoktávós hanggenerátor és egy zajgenerátor.

A C16 külsőre megegyezik a C64-gyel, csak a doboz szürke színe jelzi, hogy másik gépről van szó. A billentyűzet csoportosítása kissé megváltozott. A kurzormozgató billentyűket egymás mellé tették, ami ugyan szokatlan lesz a régi Commodore-felhasználóknak, viszont kétségteljesen célszerű módosítás. Az angol szaklap az új gép fő hiányosságaként a csatlakozókat jelölte meg. A C64-en található standard D9 joystick aljzatok helyett kisebb, kerek csatlakozókat alkalmaztak, melyekhez csak a Commodore gyártmányú Super joystick használható. Megváltozott a magnetofoncsatlakozó is, a C16-tal nem működik sem a régi 1530-as, sem a C2N szalagegység. Ez azonban nem jelent problémát, mivel az új számítógéphez magnetofont is mellékelnek. (A C16 egyébként szinte semmilyen vonatkozásban nem kompatibilis a C64-gyel.) Megszűntek a felhasználói portok, ezek szerepét a cartridge-csatlakozó vette át.

Az új típusú csatlakozók alkalmazását a cég utólag azzal indokolta, hogy a C16-ot eredetileg a legújabb Commodore Plus 4 géppel azonos kivitelűre tervezték. Ez a gép viszont kisebb burkolatot kapott, aminek a

hátlapján nem fértek el a nagyobb, régi csatlakozók. A C16-ot később ugyan nagyobb dobozba szerelték, de nem lett volna ésszerű egyidőben két gépet különböző csatlakozókkal piacra dobni.

Hasznos szolgáltatás a RESET gomb, ami már nem a megszakításkérését használja, hanem közvetlenül a processzorral van összekötve. Lenyomása az összes Basic-mutatót nullázza. Látszólag tehát elvész a RAM-ban levő Basic program, valójában viszont továbbra is a memóriában marad és egyszerűen visszanyerhető. Sajnos a tápegység még mindig a gépen kívül helyezkedik el, bár a készülékben elegendő hely állt volna rendelkezésre.

A kisebb-nagyobb hardverhiányosságokat ellensúlyozza a kitűnő Basic interpreter. A Commodore gépeknek mindig a Basic volt a gyenge pontjuk, azonban a C16 kivívta a PCT elismerését. A grafikai utasítások célszerűek és van is belőlük szép számmal. A felhasználónak 16 szín 8-8 árnyalata áll rendelkezésére. A megszokott grafikai utasításokon kívül néhány különleges is fellelhető. A C16-nak nincs hardver sprite kezelése, ezt szoftver úton végzi. (Sajnos emiatt lassú, tehát nem használható nagyobb alakok mozgására, holott épp erre szánták.) Vannak diskkezelő utasításai, a Basic-ban kevésbé elterjedt, ám célszerű ciklusképző szerkezetei (DO...WHILE és DO...UNTIL) és formátumos outputja. Lehetséges a decimális-hexadecimális konverzió, ha pedig valaki szövegkezelő programokat akar írni, szokatlanul nagyszámú stringműveletben válogathat. A programozó munkáját segíti néhány olyan szolgáltatás, mint az újrastartás, vagy a nyomkövető funkció. A C16-nak van beépített monitorprogramja is, mely eltérően az eddigi gyakorlattól egy assemblert is tartalmaz.

A gépet a már jócskán elavult VIC kiváltására tervezték, de úgy tűnik – dacára a kisebb memóriának – méltó versenypartnere lesz elődjének, a C64-nek is. Bár a joystick csatlakozók megváltoztatása elég kellemetlenül érintette a játékipacot, a PCT mégis ezen a téren jóslol nagy jövőt a C16-nak. Ezt látszik alátámasztani, hogy jó néhány szoftverház kezdte meg a játéprogramok kidolgozását – a szükséges szoftver-háttér tehát hamarosan összeáll. A kezdőcsomag ára (ebben benne van a szalagegység is) Angliában 139,95 font, ami végre a Spectruméhoz hasonló alacsony ár.

Optikai lemezek!

A Colorado Springs-i Information Storage Inc. áprilisra tervezi optikai úton olvasható lemezegységeinek megjelenését. Ezeket az egységeket „egyszer írják”, „sokszor olvasák”. A lemezmeghajtó egység cserélhető 100 megabyte-os 5 1/4 colos lemez cartridge-eket használ. A lemezmeghajtó egységek ára 600 dollár körül lesz, a lemezcsonomok ára 40 dollár körül várható.

Képszepek

A NEC elektronikai cég bejelentette az új μ PD7281D chipjét, amely az első nem Neumann-féle, egyetlen chipen létező képfeldolgozó processzor. A 7281-es 5 millió műveletet hajt végre másodpercenként. Több ilyen processzor összekapcsolásával növelhető ez a teljesítmény.

Tokióból érkezett

Tokióban bejelentették, hogy a japán külkereskedelmi és ipari minisztérium feloldotta az eddigi exporttilalmat a személyi számítógépek egyes fajtáinak a szocialista országokba történő eladására. A szigetország legnagyobb számítógépgyártója, a Fujitsu pedig közölte, hogy új számítógépe gyorsabb, mint a világ bármelyik komputere. Az új szuperberendezés másodpercenként egymillió művelet végrehajtására képes. Ugyancsak tokiói hír, hogy Teleport néven egy távközlési műholdhoz kapcsolódó nagy információfeldolgozó és -továbbító rendszert helyeztek üzembe. A Teleport földi bázisa számítógéprendszerből és adó-vevő antennák együtteséből áll.

Ime, ígérünkhez híven, közreadunk egyet a HT-GÉPNYERŐ első fordulójára (zseb-számológép-szimuláció) érkezett pályaművek közül. Nem úgy, mint egyértelműen a legjobbnak tartott programot, hanem mint a legjobb programok közül a legrövidebbet, legelegánsabbat. S kísérőül híven közöljük a programhoz mellékelt rövid leírást is, egyrészt hogy segítsen a program megértésében és használatában, másrészt, hogy példát mutassunk sok szakkörnek: körülbelül ilyen leírással illik egy programot valahová elküldeni (vagy persze még részletesebbel). Ugyanis a szakkörök fele egyáltalán nem küldött a programhoz leírást! Kivételesen elnéztük az ilyen (a feladat kiírásába is ütköző) rendellenességeket, de ígérjük, hogy ha máskor is lesz ilyen jellegű pályázat, sokkal szigorúbbak leszünk!

S most következzen a budapesti Piarista Gimnázium pályázati anyaga:

A számológép-szimuláló program ismertetése

1. A program használata

A program egy „lengyel módszerrel” működő számológépet szimulál. Ez a módszer azt jelenti, hogy a beírt szám az akkumulátorba került, onnan vagy a stackbe vagy a memóriába jut, vagy fölülíródik: a bináris művelet a stack tetején lévő és az akkumulátorban lévő számmal végződik, az unáris művelet pedig az akkumulátorban lévővel. Az eredmény az akkumulátorba kerül, és ez íródik ki a displayre.

Figyelem: a „lengyel módszerrel” dolgozó kalkulátorokban nincs egyenlőségjel! (Pl. a Hewlett-Packard gépek.)

Példa bináris művelet elvégzésre:

5+4 kiszámítása: 5 EN(=NEW Line) 4+

Unáris műveletre:

$\sqrt{2}$ kiszámítása: 2 SQR (=V)

A gépnek 10 memóriája van; a memóriakezelő utasítás után beütött egyjegyű szám jelzi, hogy melyik memóriával dolgozunk.

```
1 'PIARISTA GIMNAZIUM, BP.
  A SZAKKÖR, 1. FELADAT
  HT-1080Z, ELSŐ KIADÁS
2 GOSUB 2000
10 CLS: CLEAR 500: DEFINTB-V: DEFDBLA,P,W
20 MAX=50:P2=1.5707963#: C0$="0123456789./% "+CHR$(31)+CHR$(13)+"PRXISCTGLV+-*:[O
>X%": DP=270: IP=DP+64: MP=IP+4: EP=416: D$=CHR$(202)
30 DIMW(MAX),WM(9),S$(1),IN$(1),E$(9),M$(4): S$(0)="-": IN$(0)="" : IN$(1)="INV"
40 FORI=1TO9: READE$(I): NEXTI: READE$(1),M$(2),M$(3),M$(4)
50 FORI=0TO4: PRINT,"#";I;"#";I+5;"#": NEXTI
60 PRINT,"SP=","(SP)=": PRINT2262,"HT-35": PRINT2326,"CALC": PRINT2458,"MADE IN H
UNGARY"
70 FORI=9TO52: SET(I,10): SET(I,19): SET(I,45): NEXTI: FORI=10TO45: SET(9,I): SET(52,I)
: IFI(19,SET(24,I)
80 NEXTI
90 FORI=0TO6: J=64*I+510: READE$,Q$: PRINT2J,P$: PRINT2J+33,Q$: NEXTI: GOT0300
100 DATA DIVISION BY ZERO,X<0 IN SQR,X<0 IN LOG,ILLEGAL TG,ILLEGAL INV SIN/INV
COS,ILLEGAL BASE,STACK EMPTY,STACK FULL,OVERFLOW,CLR#,STOW,RCL#,EXCH
110 DATA 1/X EXS EN OFF ON,R X NL SPC CLR
120 DATA INV sin cos tg CE,I S C T @
130 DATA PI SQR lg ln [,P V G L [
140 DATA "CLR# 7 8 9 :","0 7 8 9 :
150 DATA STOW 4 5 6 *,> 4 5 6 *
160 DATA RCL# 1 2 3 -,< 1 2 3 -
170 DATA EXCH 0 , +,- +,% 0 , / +
200 ' C$=INKEY$, C=CODE(C$)
220 C$=INKEY$: IFC$="" THEN220
230 FORC=1TO35: IFC$=MID$(C$,C,1): RETURN
240 NEXTC: GOT0220
250 ' ERROR
260 PRINT2DP," Error ";E$: PRINT2EP,E$(E):
270 GOSUB200: IFC140R15<THEN270ELSE360
300 ' FDCIKLUS
310 SP=0: AC=0: GOT0940
320 AC=0
330 IN=0
340 GOSUB600: GOSUB200: IFC=1THEN320
350 IFC12THEN500
360 IFC21,ONC-11GOT0460,320,390,310,410,420,430,440,450
370 IFC27THEN700
380 IFC32THEN800ELSE900
390 PRINT2DP,CHR$(202): FORI=0TO9: WM(I)=0: NEXTI
395 GOSUB200: IFC<>15THEN395ELSE310
400 ' SPECIALIS FUNKCIOK
410 IFSP=MAX,E=8: GOT0250ELSESP=SP+1: W(SP)=AC: GOT0330
420 AC=3.1415927#: GOT0330
430 IFAC=0,E=1: GOT0250ELSEAC=1#/AC: GOT0330
440 IFSP=0,E=7: GOT0250ELSEW=W(SP): W(SP)=AC: AC=W: GOT0330
450 IN=1-IN: GOT0340
460 AC=-AC: GOT0330
500 ' IN ACCU
510 FP=0: SA=1: IN=0: A$="" : GOT0535
520 GOSUB650: IFC12,AC=VAL(S$(A$)+A$): GOT0350
530 GOSUB200
535 IFC=11,IFFPTHEN530ELSEFP=1
540 IFC12,IFLEN(A$)=8+FPTHEN530ELSEA$=A$+C$
550 IFC12,SA=1-SA
560 GOT0520
600 ' DISPLAY
610 PRINT2EP,CHR$(220): PRINT2355,SP: PRINT2373,D$: PRINT2373,W(SP): SA=1: A$=STR
$(AC)
620 IFAC=>1D8ORAC=-1D8,E=9: GOT0250
625 IF-1D-8<ACANDAC1D-8,AC=0
630 IFLEFT$(RIGHT$(A$,4),1)="" THEN D$=LEFT$(A$,1)+". "+STRING$(VAL(RIGHT$(A$,2))-1,"
0")+MID$(A$,2,1)+MID$(A$,4): A$=LEFT$(A$,LEN(A$)-3)
640 A$=LEFT$(A$,10): AC=VAL(A$): IFRIGHT$(A$,1)="" THEN A$=LEFT$(A$,9)
650 PRINT2IP,IN$(IN): PRINT2DP,RIGHT$(
"+S$(SA)+A$,10): RETURN
700 ' SZIMPLAPONTOS FUGGVENYEK
710 X=CSNG(AC): ONC-20+6*INGOSUB720,730,740,750,760,770,725,735,745,755,765,775:A
C=CDBL(Y): GOT0330
```


Zsebszámológép a HT-n

Ilyen utasítások:

CLR # (=0) – törlés

STO # (= >) – tárolás

RCL # (= <) – előhívás

EXC # (= %) – az akkumulátor (kijelző) értékével cseréli a tartalmat

A stackben 50 hely van. A stackkel kapcsolatos utasítások:

(nyomógombok)

EN (=New Line) – az akkumulátor tartalmát beírja a stack tetejére (a többi eggyel mélyebbre teszi)

EXS (=X) – felcseréli a stack tetején lévő számot az akkumulátorban lévővel

A számológépet az OFF (=Space) utasítással lehet kikapcsolni, ezzel törlődik az összes adat; bekapcsolás az ON (=Clear) billentyűvel történik. A CE (=C) utasítás az akkumulátort törli. A függvények inverzét az INV (=I) utasítással képezhetjük a függvény billentyűjének lenyomása előtt.

A felhasználás megkönnyítése érdekében szerepel a programban is ez az útmutató.

2. A program felépítése

A program a (végére írt) használati útmutatóval kezdődik.

10–170 sorokban a változók definiálása, képernyő rajzoltatása szerepel:

300–400 sorokban a szimulációt vezérlő főciklus található, innen van elágazás az egyes speciális feladatokat végző szubrutinokba, amelyek a következők:

200–240 felhasználó közléseinek értékelése (billentyűzet)

250–270 hibajelzések

400–450 speciális funkciók

500–560 akkumulátorkézelés

600–650 képernyőkezelés

700–775 unáris műveletek (függvények)

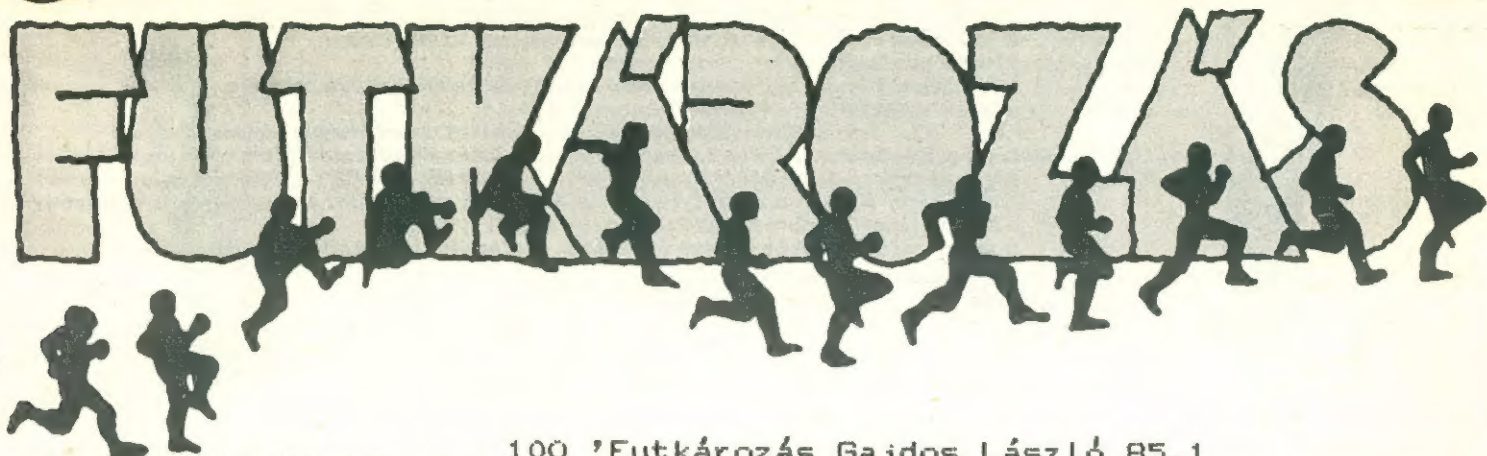
800–890 binér műveletek

900–990 memóriakezelés

```

720 Y=SIN(X):RETURN
725 IFABS(X)>1,E=5:GOTO250
727 IFABS(X)>.99999,Y=SGN(X)*P2ELSEY=ATN(X/SQR(1-X*X))
728 RETURN
730 Y=COS(X):RETURN
735 GOSUB725:Y=P2-Y:RETURN
740 IFCOS(X)=0,E=4:GOTO250ELSEY=TAN(X):RETURN
745 Y=ATN(X):RETURN
750 GOSUB740:Y=Y/LOG(10):RETURN
755 Y=10^X:RETURN
760 IFX>0,Y=LOG(X):RETURNELSEY=3:GOTO250
765 Y=EXP(X):RETURN
770 IFX<0,E=2:GOTO250ELSEY=SQR(X):RETURN
775 Y=X*X:RETURN
800 ' BÍNER MŰVELETEK
810 IFSP=0,E=7:GOTO250
820 ONC-26GOSUB850,860,870,880,890
830 SP=SP-1:GOTO330
850 AC=W(SP)+AC:RETURN
860 AC=W(SP)-AC:RETURN
870 AC=W(SP)*AC:RETURN
880 IFAC=0,E=1:GOTO250:ELSEAC=W(SP)/AC:RETURN
890 W=W(SP):IFW=0ANDAC>0,AC=W:RETURNELSEIFW<0,E=6:GOTO250
895 IFLOG(W)*AC>70,E=9:GOTO250ELSEAC=W/AC:RETURN
900 ' MEMORIA
910 N=C-31:PRINT&MP,M$(N);
920 GOSUB200:IFC=11THEN920ELSEIFC=11THEN940
930 M=C-1:PRINTM;:ONNGOSUB960,970,980,990
940 FORI=0TO4:J=64*I+37:PRINT&J,D$;:PRINT&J,W$(I);:PRINT&J+16,D$;:PRINT&J+16,W$(I+5);:NEXTI:PRINT&MP," ";:GOTO330
960 WM(M)=0:RETURN
970 WM(M)=AC:RETURN
980 AC=WM(M):RETURN
990 W=WM(M):WM(M)=AC:AC=W:RETURN
2000 CLS:PRINT"KER HASZNALATI UTASITAST? (I/N)"
2010 A$=INKEY$:IFA$=""THEN2010ELSEIFA$<>"I",RETURN
2030 CLS:PRINT"
A PROGRAM EGY LENGYEL MODSZERREL MUKODO KALKULATOR SZIMULAL.
EZ AZT JELENTI, HOGY A PROGRAM EGY STACKKEL DOLGOZIK, A BÍNER
MŰVELETEKET A STACK TETEJEN LEVO ERTEKKEL ES AZ AKKUMULATORRAL";
2035 PRINT"
VEGZI EL. AZ AKKUMULATORBA KERUL A BEADOTT SZAM, ES A GEP EZZEL
VEGZI AZ UNARIS MŰVELETEKET. A KIJELZORE MINDIG AZ AKKUMULATOR
ERTEKE IRODIK KI.";
2040 PRINT"
A GEPNEK 10 MEMORIAJA VAN. EZEKET A MEMORIAKEZELO UTASITAS UTAN
LEUTOTT EGYJEGYU SZAMMAL VALASZTJUK KI.";
2050 PRINT:PRINT"MEMORIAKEZELO UTASITASOK:"
2060 PRINT"CLR# CLEAR MEMORY - MEMORIA TORLES"
2070 PRINT"STO# STORE MEMORY - SZAM TAROLASA"
2080 PRINT"RCL# RECALL MEMORY - MEMORIA ELOHIVAS"
2090 PRINT"EXC# EXCHANGE - A MEMORIABAN LEVO SZAMOT FELCSERELI
A KIJELZON LEVOVEL (=AKKUMULATOR)"
2100 A$=INKEY$:IFA$=""THEN2100
2110 CLS:PRINT"STACKKEL KAPCSOLATOS UTASITASOK:"
2120 PRINT"EN : ENTER - AZ AKKUMULATOR BERA KJA A STACKBE."
2130 PRINT"EXS: EXCHANGE STACK - AZ AKKUMULATOR FELCSERELI A STACK
TETEJEN LEVO SZAMMAL"
2140 PRINT:PRINT"AZ EGYEB MŰVELETEK KET FO CSOPORTBA OSZTHATOK:"
2150 PRINT"1. UNARIS MŰVELETEK: EGY SZAMON VEGZI EL. PL. GYOKVONAS"
2159 PRINT"2. BÍNER MŰVELETEK: KET SZAMON VEGZI EL. PL. ALAPMŰVELETEK"
2160 A$=INKEY$:IFA$=""THEN2160
2161 CLS:PRINT:PRINT"AZ 5+4 OSSZEADAS TEHAT IGY VEGEZHETO EL:"
PRINT"5":PRINT"EN":PRINT"4":PRINT"+":PRINT"(KIJELZON AZ EREDMENY)"
2165 PRINT:PRINT"
KEREM, ALLITSA A GEPET SZELES KEPERNYORE.
A PAGE BILLENTYUVEL VALTHATJA A KEPET."
2170 A$=INKEY$:IFA$=""THEN2170
2180 RETURN

```

A Primóra közölt játékprogram a következőképpen működik: a képernyőn egy futkározó fejet kell vezetni, mely a billentyűzet érintésére, ill. bizonyos idő eltelté után irányt változtat vagy egyhelyben megáll. Az irányváltoztatás véletlenszerű.

A pályán akadályok és pontok vannak. Ha a bábu nagy akadályba ütközik, elveszít egy életet. Az ötödik élet elvesztése után új játék kezdődik, mely a billentyűzet érintésére indul. Ha azonban a bábu pontba ütközik, akkor azt „fölfalja” és 20 pontot nyer.

A játékot tenyérrel érdemes játszani. A Primo így biztosan érzékeli a billentyűzet érintését.

A játékban előre definiált karakterek vannak, melyek előállítását az első 7 sor végzi.

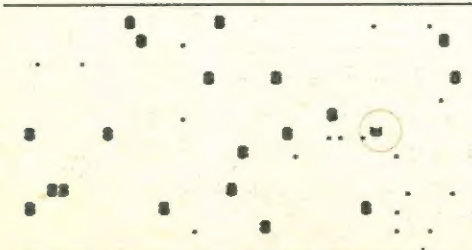
A karakterdefiníció után az 1000-es sorra történik ugrás. Ahol az alapkép kirajzolása és a változók értékeinek beállítása következik.

A 200-as és 490-es sor között van elhelyezve a program fő része. A 220-as sor végzi az irányváltoztatást, itt egy ellenőrzés is el van helyezve, ami kiküszöböli, hogy a véletlenszám-generátor az előző értékkel egyező értékeket állítson elő, mivel ebben az esetben az irányváltoztatás nem következik be.

A 360-as sor a kis pontokkal, a 370-es pedig a nagy pontokkal való ütközést ellenőrzi.

A 420-as sor biztosítja a 20. elmozdulás utáni irányváltoztatást. Az A\$-nak SPACE értéket ad, és a 280-as sorra küldi a programot. A tulajdonképpeni irányváltoztatás itt történik meg egy ellenőrzés segítségével, ami az A és B értékeket állítja újra abban az esetben, ha AS nem egyenlő üres stringgel. A programban szereplő változók: A-Y irányváltoztatás; B-X irányváltoztatás; AA/BB - A, ill. B ellenőrzése; X/Y - A bábu X, Y helyzete; P - Pontszám; LP - legmagasabb pontszám; E - életek; M - mozgás (20 elmozdulás utáni irányváltoztatáshoz). *Gajdos László*

Pontszám : 15 Eddigi csúcs : 373
Életek : 5



```
100 'Futkározás Gajdos László 85.1
110 A$="": C=0: A=0: AA=0: B=0: BB=0: X=0: Y=0: P
=0: LP=0: E=0: M=0: V=-1
120 DIM K%(18): C=VARPTR(K%(0))
130 POKE C,66,60,90,126,102,36,66,61
140 POKE C+8,28,62,62,28,62,62,62,29
145 POKE C+16,66,60,90,126,66,60,0,1
150 POKE 16459,C-256*INT(C/256),INT(C/256)
160 GOTO 190
170 IF LP<P THEN LP=P
180 IF INKEY$="" THEN 180
190 CLS
200 FOR A=0 TO 255: SET(A,0): SET(A,162): NEXT A
210 PRINT "Pontszám : 0 Eddigi csúcs :";LP
220 PRINT "Életek : 5"
230 FOR A=1 TO 20: PRINT$ RND(12)+2,RND(41),".":
NEXT A
240 FOR A=1 TO 20: PRINT$ RND(12)+2,RND(41),CHR$(
129): NEXT A
250 X=20: Y=8: PRINT$ Y,X,CHR$(6)CHR$(128): M=0:
P=0: E=5
260 IF INKEY$="" THEN 260
270 A$=INKEY$
280 IF A$<>"" THEN A=INT(RND(3)-2): B=INT(RND(3)-
2): M=0: IF A=AA AND B=BB THEN 280 ELSE BEEP 100,
10
290 AA=A: BB=B
300 PRINT$ Y,X,CHR$(6)" ";
310 Y=Y+A: X=X+B
320 IF Y<3 THEN Y=14
330 IF Y>14 THEN Y=3
340 IF X<0 THEN X=41
350 IF X>41 THEN X=0
360 IF POINT((X+1)*6,184-(Y*12))<>0 AND POINT((X+
1)*6,188-(Y*12))=0 THEN P=P+20: BEEP 200,20
370 IF POINT((X+1)*6,188-(Y*12))<>0 THEN E=E-1: B
EEP 1000,20: PRINT$ 1,11,E: IF E=0 THEN 170
380 PRINT$ Y,X,CHR$(129+V);: V=-V: P=P+1: PRINT$
0,11,CHR$(6);P
390 IF P/25=INT(P/25) THEN PRINT$ RND(12)+2,RND(4
1),CHR$(129);
400 IF P/15=INT(P/15) THEN PRINT$ RND(12)+2,RND(4
1),".":
410 BEEP 5,10: M=M+1
420 IF M=20 THEN A$="" : GOTO 280 ELSE 270
430 END 'Futkározás
```


Úgy látszik, a Primo lesz a kettős számú iskolás-mítógép. (Erre talán telik majd az általános iskoláknak is.) Ezért valószínűleg az eladott gépek számánál sokkal nagyobb a használók száma.

A géppel kapcsolatos néhány olyan „felfedezésemet” kívánom ismertetni, amelyek ismerete sok bosszúságtól kíméli meg a felhasználókat.

a) A felhasználói kézikönyv nem említi (miért?) a képernyőre nyomtatást pozicionáló utasítást, amelynek ez az alakja: PRINT m,n,l.

m jelentése: „sor”, lehet szám, változó vagy aritmetikai kifejezés. Legkisebb értéke 0, legnagyobb 15.

n jelentése: „oszlop”, lehet szám (konstans), változó vagy aritmetikai kifejezés. Legkisebb é.: 0, legnagyobb: 41.

b) A felhasználói kézikönyv nem ír a BEEP m,n utasításról (miért?). Ez a számítógép hangszóróját működteti.

m arányos a hangrezgés egy periódusának időtartamával (minél nagyobb, annál mélyebb a hang).

n arányos a hangrezgések számával (nem a másodpercenkénti rezgésszámmal!).

Tehát a hang hossza hozzávetőlegesen az m*n szorzattal arányos! Ebben különbözik a Spectrum megfelelő utasításától.

c) A képernyőre függőlegesen is lehet írni. Közvetlenül a CTR gomb és a 0 megnyomásával. Vízszintes írásba viszont a CTR+W megnyomásával lehet.

Programból pedig a PRINT CHR\$ (15) utasítással lehet ebbe az üzemmódba váltani, visszatérni pedig a PRINT CHR\$ (23)-mal lehet.

d) a 16452 memóriacím legfelső bitje (MSB) határozza meg, hogy az INKEY\$ nagybetűt vár-e vagy kisbetűt. Az UPPER gomb ezt a bitet kapcsolja be vagy ki. 1 = nagybetű, 0 = kisbetű.

e) A 16455 memóriacím a mindenkorli üzemmódot határozza meg. Ennek segítségével akár egy utasítással be lehet állítani a teljes üzemmódot. Még olyasmit is, amire BASIC-ből nincs lehetőség (pl. láthatatlan írás a képernyőre).

7. bit (MSB) 1 = nyújtott karakter, 0 = nem

6. bit 1 = aláhúzás bekapcsolva, 0 = kikapcsolva

5. bit 1 = a képernyő világos, 0 = sötét

4. bit 1 = a karakter környezete a képernyővel ellentétes („negatív alap”)

0 = azonos megvilágítású

3. bit

2. bit) együtt 11_B akkor van előtörítés, 00_B akkor nincs

2. bit előtörítés

1. bit 1 = a betű sötét, 0 = a betű világos

0. bit 1 = függőleges írás, 0 = vízszintes írás

f) Ha az előtörítés be van kapcsolva, a többi üzemmód bizonyos kombinációinál a CLS utasítás az előtörítést kikapcsolja. Ezért, ha szükséges, egy CLS után ismét be kell írni az előtörítést (CHRS(6)).

g) A Primo belső órája a következő címeken található:

16447

16446

16445

(Ez nem más, mint egy 24 bites számláló.)

A számítógép bekapcsolásakor az óra 0,0,0-ról indul, a legkisebb című gyorsan eléri a 255-öt, akkor 0-ra vált, de ugyanakkor a 16446 értéke nő eggyel. Azután, ha a 16446-os regiszter éri el a 255-öt, ez is 0-ba fordul, és a 16447-et növeli 1-gyel. 255,255,255 után ismét 0,0,0-ba fordulnak. E memóriacímekre tetszőleges értékek beírhatók POKE-kal, azután a beírt értéktől folytatják a számlálást.

Figyelem! A NEW utasításra a belső órával semmi sem történik (folytatja a számlálást), a RESET gomb megnyomása lenullázza!

h) A PI azonosító értéke állandóan 3.14159 (π értéke 7 tizedesig). Ha valakinek tényleg nagy szüksége van egy szabad PI azonosítóra, írhatja így: P I, de akkor mindig így kell írnia, bármilyen értéket adhat neki. Ez bemutatja a Primo szokatlan nagyvonalúságát: azonosítók belsejében

szóközöket hagyhat a programozó, és az teljesen azonosnak veszi a szóköz nélküli azonosítókkal. (Kivéve, ha az egyébként kulcsszó lenne. L. PI esete.) Sőt! Program-sorszám belsejében is szabad szóközöt hagyni, a szóközöket figyelmen kívül hagyja a gép.

Tudjuk, hogy általában nem lehet kulcsszavakat azonosítónak használni. A Primo esetében ez megkerülhető úgy, hogy az azonosító belsejébe egy szóközt írunk.

Tehát

XA ugyanaz, mint X A, AA ugyanaz, mint A A,

OR kulcsszó, de O R azonosító,

TO kulcsszó, de T O azonosító,

PRINT kulcsszó, és P RINT szintaktikai hiba, mert tartalmazza az INT kulcsszót!

i) A CLEAR vagy CLEAR n utasításnak van egy mellékhatása is. A CLEAR utasítás, mint ismeretes, törli az összes változót, a CLEAR n pedig még n byte helyet is lefoglal a szöveges változóknak.

Az ON ERROR GOTO n utasítás hatását a megszokott módon ON ERROR GOTO 0 utasítással lehet megszüntetni. Sajnos azonban megteszi ezt a CLEAR mindkét változata. (Bár ERR és ERL megtartja az értékét!)

j) A CLEAR kellemetlenkedik másképpen is. Ha pl. CLEAR 1000-rel lefoglaltunk ezer byte-ot, ez a lefoglalt stringmemóriarész megmarad a NEW vagy akár a RESET gomb megnyomása után is.

Koszper Vilmos

Tisztelet szerkesztőség!

Nemrégiben kezembe került a BIT-LET februári száma, amelyben a BENCHMARK programokat közölték. Mivel éppen volt nálam egy Primo néhány napos használatra, beírtam ezeket a programokat.

Szeretném közölni a tesztek eredményét:

BM1: 1,6 s BM5: 21,1 s

BM2: 7,7 s BM6: 33,6 s

BM3: 18,5 s BM7: 51,6 s

BM4: 19,2 s BM8: 82,9 s

A Primo – gyorsaságát tekintve – tehát a középosztályba sorolható.

Vámos György

MEGJELENT

A



**BASIC
BAJNOKSÁG**

**SINCLAIR SPECTRUM
COMMODORE 64
4T1080Z
M08X**

számítógépekre írt
programfeladatok
és megoldások
gyűjteménye

139 oldalon, 60 Ft-ért
KAPHATÓ

az Akadémia Könyvesboltban (V. Váci u. 22.)

a Műszaki Könyvklubban (VI. Bajcsy-Zsilinszky u. 22.)

(utánvétellel is megrendelhető)

a Statisztikai és Számítástechnikai Könyvesboltban

(II. Keleti Károly u. 10.)





SZAKKÖRÖKNEK!

LEBEGŐPONTOS ARITMETIKA A SPECTRUMON II

Halász Péter írásának első részét januári BIT-LET-ünkben közöltük.

Most már ideje, hogy megnézzük, hogyan végeztethetjük el a géppel a kívánt műveleteket egy gépi kódú program során. A lebegőpontos kalkulátort egy RST 28 H utasítással lehet meghívni. Az elvégzettetni kívánt műveleteket pedig közvetlenül a hívás helye után álló byte-okkal lehet megadni.

A legfontosabb kódok jelentése:

kód (hexa)	művelet
01	megcseréli a két utolsó számot a stackben
02	törli az utolsó számot a stackből
31	az utolsó egységet még egyszer beírja, így az kétszer fog szerepelni
A0	a 0 konstans a stack végéhez írja
A1	az 1 konstans a stack végéhez írja
A2	a 1/2 konstans a stack végéhez írja
A3	A $\pi/2$ konstans a stack végéhez írja
A4	a 10 konstans a stack végéhez írja
Cn	az n-edik ($0 \leq n \leq 5$) memóriába írja a stack végén álló számot (a stackből nem törli!)
En	az n-edik ($0 \leq n \leq 5$) memóriában tárolt számot a stack végéhez írja

(a következő műveleteknél a két utolsó egység törlődik, és a művelet eredménye a stack végére íródik)

kód (hexa)	művelet
03	kivonás
04	szorzás
05	osztás
06	hatványozás
0F	összeadás
17	string-összeadás, azaz a megjelölt stringeket egymás után fűzi

(A következő műveletek az utolsó egységet törlik, és a művelet eredményét a helyére írják. Mivel ezek egy kivétellel pontosan megfelelnek egyes BASIC függvényeknek, ezért ezek nevét írjuk le.)

kód (hexa)	művelet				
18	VAL\$	22	ASN	2A	ABS
1B	ellentettképzés	23	ACS	2B	PEEK
1C	CODE	24	ATN	2C	IN
1D	VAL	25	LN	2D	USR
1E	LEN	26	EXP	2E	STR\$
1F	SIN	27	INT	2F	CHR\$
20	COS	28	SQR		
21	TAN	29	SGN		

Ezen kódsorozat végét minden esetben egy 38H kód jelzi. Ennek hatására a vezérlés visszaadódik a hívó programnak, a 38H-t követő helyre.

Az így kódolt műveletek végzése során keletkező hibákat a gép ugyanazokkal a hibajelzésekkel jelzi, mintha ezeket BASIC-ben végeztetnénk.

A kalkulátor használata során a következőkre kell még ügyelni:

1. a stackben ne maradjon „szemét”, mert sokszori hívás esetén a munkaterület megtelhet;
2. egyes funkciók használják a 0–3 memóriákat; a karakterkiíró rutin a 0-ást és az 1-est. Ezek tehát törlődhetnek. A fentiek alapján (a stack végén álló számot X-nek nevezve) SIN X+COS X kiszámítása a következő assembler rutinnal történhet (az eredményt a stack végén kapjuk):

RST	28H
DEFB	C5H
DEFB	1FH
DEFB	E5H
DEFB	20H
DEFB	0FH
DEFB	38H

HT-1080Z
SZÁMÍTÓGÉPHEZ

PUSKA

Puskának neveztük el azt a BASIC összefoglalót, amelyet Rohonyi Pál és Király László a soproni Erdészeti és Faipari Egyetem számítástechnikával foglalkozó tanárai – lapunk olvasói – készítettek és küldtek el. A puska a HT iskolaszámítógéphez készült és igyekszik összefoglalni egy lapon mindazokat a tudnivalókat, amelyekre a gép használójának szüksége lehet programozás közben. Az efféle praktikus összefoglaló táblázatoknak, kártyáknak nagy keletje van a mikrogépesek táborában. Így hát ha eme első puskánk megihleti valamelyik olvasónkat, s hasonló összefoglalót készít más olyan gépekhez, amelyek tudnivalóit még ez ideig nem foglalták össze, hozzáférhető helyen közölt táblázatban, szívesen állunk elébe, s szívesen közöljük, ha eljuttatja hozzánk. Ha pedig itt közölt puskánkban körültekintő ellenőrzésünk ellenére is hibát talál olvasónk, kérjük, írja meg.

ARITMETIKAI MŰVELETI JELEK

+, -, *, /, —
<, <=, >, >=, =, <>

LOGIKAI MŰVELETEK

AND, OR, NOT

Eredmény: — 1, ha igaz a feltétel
0, ha nem igaz a feltétel

STRING MŰVELET

sk1+sk2 összekapcsolás

MŰVELETEK SORRENDJE

Egy kategórián belül balról jobbra

1. zárójel
2. belső függvény
3. hatványozás
4. előjelváltás
5. szorzás, osztás
6. összeadás, kivonás
7. relációk
8. logikai műveletek a) NOT
b) AND
c) OR

RÖVIDÍTÉSEK

a aritmetikai n név
c constans s string
e egyszerű u utasításszám
i integer (egész) v változó
k kifejezés

α egy betű (A, B, ..., Z)

... rövidítés(ek)-nek megfelelő lista, (elválasztójelként a vessző adható meg)

() utasításhoz tartozó zárójel, ki kell írni

[] a szögletes zárójelben levő paraméter használata nem kötelező

{ } a kapcsos zárójel egyik sorát ki kell választani

[] összetartozó utasítások

[CLEAR] kezelőgomb

VÁLTOZÓK TÍPUSÁNAK KIJELÖLÉSE

i=0,1,...,9

Változó (vn) alapján	Típusa	Pontossága	Definiálás alapján
$\alpha \left[\left\{ \begin{smallmatrix} \alpha \\ i \end{smallmatrix} \right\} \right] [!]$	egyszeres pont.	max. 6 számj.	DEFSNG
$\alpha \left[\left\{ \begin{smallmatrix} \alpha \\ i \end{smallmatrix} \right\} \right] \%$	egész típusú	—32768 → 32767	DEFINT
$\alpha \left[\left\{ \begin{smallmatrix} \alpha \\ i \end{smallmatrix} \right\} \right] \#$	dupla pont.	max. 16 számj.	DEFDBL
$\alpha \left[\left\{ \begin{smallmatrix} \alpha \\ i \end{smallmatrix} \right\} \right] \$$	karaktársorozat	max. 255 jel	DEFSTR

A név szerinti kijelölés a DEF $\alpha\alpha\alpha$ típusú meghatározást felülírja

PARANCS ÜZEMMÓD

([NEW LINE]-ra végrehajtódik)

NEW

tároló törlése

[CLEAR]

képernyő tartalmának törlése

AUTO [U, Δu]

automatikus sorszámozás

RE [u] [Δu]

újra sorszámozza a programot

CLEAR [i]

i byte-ot töröl string részére + törli a változókat

LIST [u1] [—u2]

program listázása

LLIST [u1] [—u2]

nyomtatóra listáz

DELETE [u1] [—u2]

programsor (sorok) törlése

EDIT {u}

szövegszerkesztő üzemmódba áttérés

RUN [u]

program futtatása

[BREAK]

programvégrehajtás megszakítása

CONT

[BREAK] és programközbeni STOP

TRON

követés funkciót bekapcsolja

TROFF

követés funkciót kikapcsolja

CSAVE [# — { 1 } 2 }] sc

kazettára írás

CLOAD? [sc]

programtárolás ellenőrzése

CLOAD [# — { 1 } 2 }] sc

kazettáról olvasás

SYSTEM

motnior üzemmódba áttérés

BELSŐ ARITMETIKAI FÜGGVÉNYEK

x=ak, szögek esetén radián

Átszámítások: rad ← fok · π/180; fok ← rad · 180/π

π = 3,14159265358979 e = 2,71828182845905

SIN(x) sin x LOG(x) ln x

COS(x) cos x EXP(x) e^x

TAN(x) tg x ABS(x) |x|

ATN(x) arc tg x SQR(x) √x

FIX(x) csonkít

INT(x) x egész részét adja (az eredmény valós)

CINT(x) x egész részét adja (az eredmény egész típusú)

CSNG(x) egyszeres pontosságúvá alakít

CDBL(x) kétszeres pontosságúvá alakít

SGN(x) előjel függvény: —1, ha x < 0; 0, ha x = 0;

1, ha x > 0

RANDOM az RND(x) függvény véletlenszámtól kezdődik

RND(x) x = 0: egyszeres pontosságú véletlenszámot ad a

(0, 1) intervallumban

x = i: 1 és i közötti egész számot állít elő

STRINGEKEL KAPCSOLATOS FÜGGVÉNYEK

ASC(sk) első karakter ASCII kódját adja

CHR\$(ak) ASCII kód alapján karaktert ad

LEN(sk) string hosszát adja

LEFT\$(sk, ak) string bal oldali része

MID\$(sk, ak1, ak2) string közepéből vág ki

RIGHT\$(sk, ak) string jobb oldali része

STRING\$(i, {sc
ASCII kód}) adott karakterből i hosszúságú stringet állít elő (i=0, ..., 255)

STR\$(ak) stringgé alakít

VAL(sk) numerikus értéké alakít

GRAFIKUS FUNKCIÓK

x=ak1, sorok száma 0, . . . , 47
y=ak2, oszlopok száma 0, . . . , 127
CLS a képernyő tartalmának törlése
SET(x, y) fehér téglalap kerül az x,y helyre
RESET(x,y) a fehér téglalapot törli az x,y helyen
POINT(x,y) a függvény értéke -1, ha az x,y helyen van fehér téglalap; 0, ha nincs

SPECIÁLIS FUNKCIÓK

i=0, . . . , 32767(65535)
i1=15360, . . . , 32767 (65535)
i2=0, . . . , 255
PEEK(i) i-edik byte értékét adja
POKE i1, i2 i1-edik byte-ba i2 értékét írja
MEM memória szabad byte-jainak száma
ERL hibás sor számát tartalmazza
ERR ERR=(hibakód-1)*2
INKEY\$ billentyűzetről egy jelet fogad
VARPTR(evn) változó értékének címét adja
egész típus változó: LSB, MSB
egysz. pont. változó: LSB; MSB1, MSB2, kitevő
kétszeres pont változó: LSB, MSB1, . . . , MSB7, kitevő
string változó: hosszúság, kezdőcím, LSB, MSB

MONITOR ÜZEMMÓD

i, i1, i2=4000, . . . , 7FFF
(FFFF) hex. cím
/12710 CPU regisztereinek aktuális tartalma
R CPU regiszterek módosítása (x továbbugrás)
Mi i-edik memória módosítása (x kiugrás)
Di memóriatartalom megjelenítése (x visszatérés)
B visszatérés BASIC-be
G i1[, i2] program futtatása i1 címtől, i2 a töréspont
/12294 kisbetű (SHIFT)-tel
/12299 ———— ,RE parancs
/12288 ———— , villogó cursor

SZÖVEGSZERKESZTŐ ÜZEMMÓD

EDIT {u} aktuális, ill. u-adik sor javítható
NEW LINE tárol minden változtatást
[i] SPACE kurzor i-vel jobbra vitele
[i] ← kurzor i-vel balra vitele
SHIFT ↑ befejezi a közbeírási parancsot
H jobbra eső rész törlése, beírás követk.
I beszúrás (jobbra tolással)
X a sor végére állás + beszúrás
L sor kilistázása
A töröl minden változtatást
E tárolja a változtatásokat
Q vissza a parancs üzemmódba a változások tárolása nélkül
[i]D i karaktert töröl
[i]C i karaktert cserél
[i]Sc i-edik c karakter elé állás
[i]Kc i-edik c karakterig töröl

HIBAÜZENETEK

1 NF FOR nélküli NEXT	13 TM típuskeveredés
2 SN szintaktikai hiba	14 OS stringterület kicsi
3 RF GOSUB nélküli RETURN	15 LS a string túl hosszú
4 OD nincs több adat	16 ST túl összetett stringműv.
5 FC hibás függvényutasítás	17 CN CONT-tal nem folytat- ható
6 OV túlcsordulás	18 NR NO RESUME (pr. vége)
7 OM nincs több tárolóhely	19 RW RESUME, ERROR nélkül
8 UL nem létező sor	20 UE nem kírható hiba
9 BS dimenzió túl nagy	21 MD oparandus hiányzik
10 DD már dimenzionált	22 FD hiányos file
11 /0 nullával osztás	
12 ID hibás felhasználás	

BASIC-UTASÍTÁSOK

ÉRTÉKADÁS

[LET] {av=ak}
 {sv=sk}

TÖMB MÉRTEZÉSE

DIM vn(i1 [,i2, . . . , in])

BE- ÉS KIÍRATÁSI UTASÍTÁSOK

PRINT [TAB (ak)][k][{.}]. . . képernyőre írás
LPRINT ———— sornymutatóval ír
PRINT @ i, k. . . a képernyőn elfér 16 sor és
64 oszlop; i=0, . . . , 1023
PRINT USING sv, k kírás formátum szerint
INPUT [sc:] v. . . adatbevitel billentyűzetről
DATA c. . . adatok tárolása a pr.-ban
READ v. . . olvasás a DATA-ról
RESTORE DATA elejére állítás
PRINT # — {1}
 {2}.k. . . kazettára írás
INPUT # — {1}
 {2}.v. . . kazettáról beolvasás
REM megjegyz. elhely. a pr.-ban

VÉGREHAJTÁS VEZÉRLÉSE

GOTO u ugró utasítás
ON ak GOTO u1, u2, . . . elágazó ugró utasítás
GOSUB u alprogramra ugrás
ON ak GOSUB u1, u2, . . . elágazó alprogr. ugrás
RETURN alprogr. vége, visszatérés
IF k {THEN u1
 [THEN] utasítás} {ELSE {u2
 utasítás}} feltételes utasítás
FOR aev=ak1 TO ak2 [STEP ak3]
NEXT [aev] ciklusszervezés
ON ERROR GOTO u hiba esetén ugrik
RESUME {u
 NEXT} pr. végrehajtás folytat
STOP programfutás megállít.
END programfutás vége

TAVALYI HÓ



A BYTE című lap tavalyi decemberi számának szerkesztőségi cikke összefoglalást ad a mikrogépek területének tavalyi, nevezetesebb eseményeiről. A teljesség igénye nélkül érdemes nekünk is visszatekinteni arra, hogy melyik szakterületen mi is történik az 1984-es évben.

Az év legjelentősebb új mikroszámítógépei

- Macintosh
- IBM PCjr
- IBM PC AT
- Hewlett-Packard 150
- Hewlett-Packard hordozható kivitel
- Tandy 2000, 1200 és 1000
- Data General/One
- Apple IIc
- AT&T PC
- Corona Mega PC
- Tektronix 4404 Artificial Intelligence System (Mesterséges Intelligencia Rendszer)
- NEC APC III
- Seiko karóra-számítógép

Újdonságok a mikroszoftver területén:

- az operációs rendszerek területén: számos új UNIX rendszer született, a ProDOS operációs rendszer meghódította az Apple II világot, bejelentették az MS-DOS 3.0 rendszert. Született jó néhány konkurens operációs rendszer változat (CP/M, DOS stb.);

- a programnyelvek, illetve fordítóprogramok területén a Basicnek és a C programnyelvnek sok új változata született. Megjelent a microPROLOG és a Modula 2-nek néhány változata. Nevezetes esemény volt a Turbo-Pascal megjelenése.
- a mesterséges intelligencia területén megjelent termékek:
 - a Teknowledge M.1-e,
 - a General Research TIMM nevű „szakértő” rendszere,
 - az Expert-Ease
 - a Level 5 Research Insight Knowledge rendszere
- az alkalmazási rendszerek területén a bőség zavara uralkodott, beleértve az IBM Top-View és a Digital Research GEM-jét. Megjelentek a „gondolkodást”, tervezést segítő eszközök, mint a ThinkTank vagy a MaxThink. Komoly fejlesztések történtek a felhasználóra orientált adatbáziskezelő rendszerek területén: megjelent az Infoscope, a Please 4-1-1 és a Fas Fact. A régen várt dBASE III is elkészült, csakhogy, mint a versenytársa az R:base 4000. Az információviisszakereső rendszerek kö-

zül a CLOUT és az In-Search érdemel említést. Integrált alkalmazási rendszerek a Symphony, a Framework és a PayFlo voltak. A kommunikációs alkalmazások esetében újdonság volt a Fido „hirdető tábla” és a FidoNet rendszer.

Új termékek

a kiegészítő hardver területén:

- nagy jelentőségű volt a 1200 bpi-s modemek árának nagymértékű árcsökkenése, hiszen ez lokést adott a hálózatok alkalmazásának,
- a CMOS technológia nagy sebességgel terjedt és ez a tény sok gép méretét és melegeledését csökkentette,
- a folyékony kristályos képernyők méretét sikerült 25 sorra és 80 oszlopra növelni,
- a RAM memóriák mérete 256 K-ra nőtt,
- a mikroprocesszorok mérete egyre inkább 32 bitesre nőtt,
- az „egér” használata általánossá vált,
- a háttértárolók területén megjelentek a viszonylag olcsó optikai tároló eszközök, a Winchester technológiával készült „kemény” háttértárak kompaktabbak és olcsóbbak lettek,
- a sornyomatók között a Hewlett-Packard jóvoltából elérhető áron jelentek meg a lézer sornyomatók.

HARDVER ÖTLETEK

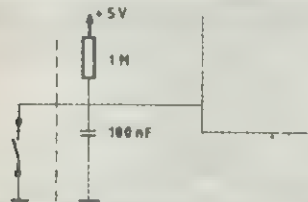


A számítógépek helyes működéséhez – többek közt – szükséges, hogy egy-egy adott memóriacímen vagy regiszterben egy állandó vagy megfelelő módon változó, de mindenképpen meghatározott tartalom legyen. A gépekben felhasznált integrált áramkörök egy része a tápfeszültség bekapcsolásakor előre meghatározhatatlan állapotba áll be, például a mikroprocesszor regisztereinek vagy az írható-olvasható memóriáknak (RAM) a tartalma véletlenszerű értékeket vesz fel. Ezért szükség van a számítógépek bekapcsolása után egy alapállapotba hozásra, amit megfelelő áramköri kialakítással automatikusan érnek el. A mikroprocesszorok és az egyéb bonyolult funkciókat ellátó áramkörök többnyire el vannak látva egy megfelelő kivezetéssel (RESET, RES, CLEAR stb), aminek aktiválásával az áramkör alaphelyzetbe hozható. A COMMODORE gépekben ilyen a mikroprocesszor, a perifériákat csatoló integrált áramkörök, valamint a C 64-ben a hangot előállító integrált áramkör.

Valószínűleg mindenki – aki dolgozott számítógépen – tapasztalta már, hogy egy-egy hibás program futtatása a gép lemerevedéséhez vagy valamilyen hibás működéséhez vezetett. Ezek a hibák többnyire a fentebb említett meghatározott tartalmak valamilyen „elrontásából” (átírásából) erednek, és sokszor csak a számítógép alapállapotba hozásával szüntethetők meg.

A COMMODORE gépeken az alapállapotba hozás a STOP-RESTORE kombináció megnyomásával mindaddig elérhető, amíg a 790–791-es (hexa 0316–0317) memóriacímen levő tartalmat szándékosan vagy véletlenül nem írjuk át. Utána már csak a gép ki-be kapcsolása segít, amikor – ellentétben a STOP-RESTORE használatával – a memóriában levő program is elvész, és mindent elölről lehet kezdeni. Mivel nem csak hobbi, hanem munkám is a számítástechnika, dolgoztam már komolyabb számítógépeken is, és számomra rendkívül idegen és bosszantó volt, hogy az otthoni VC 20-at időnként csak a gép ki-be kapcsolásával tudtam észhez téríteni. Ezért a saját gépembe, valamint egyik barátom C 64-es gépébe is építettem RESET kapcsolót, amivel a gép bármilyen helyzetből a bekapcsolás utáni állapotra hozható, de a memóriában levő program semvész el.

A módosítás a bekapcsolás utáni automatikus RESET impulzust előállító áramkörhöz illeszkedik, mindössze egy pillanatkapcsolóra és kevés vezetékre van szükség az elkészítéséhez. A bekapcsolás utáni RESET impulzust a VC 20 és VIC 20 gépekben egy NE555, a C 64-ben egy NE655 típusú integrált áramkör végzi. (Az NE555 egy tokban két NE555-nek megfelelő áramkört tartalmaz, a másik fele a C 64-ben a STOP-RESTORE-hoz van felhasználva.) Az eredeti áramkör kialakításának lényege, hogy egy ellenállásból és kondenzátorból álló – az NE555 illetve NE556-ra kötött – időzítő tag megfelelő késleltetést ad. A tápfeszültség állandósulása után is megfelelő feszültség szinten (0 Volt) tartja a mikroprocesszort és a többi integrált áramkör RESET kivezetését. A kondenzátornak egy pillanatkapcsolóval történő áthidalásával bármikor előidézhetünk RESET impulzust. A kapcsolót a gép hátoldalán célszerű elhelyezni, hogy a véletlen megnyomást elkerüljük.



Az integrált áramkörök típuszáma eltérhet. (A módosított gépekben NE555 helyett UA555, NE556 helyett MC3456 volt.) Segíthet az azonosításban az integrált áramkör pozíciószáma, ez a panelre is fel van festve. A C 64-nél U20, VC 20-nál UB6 volt, a VIC 20 rajzán UE6-os jelölést láttam.

A programok RESET utáni újraindítását kétféle módon lehet biztosítani. Az egyik módszer lépcsői:

- A gép a felhasználó rendelkezésére álló memória (BASIC terület) kezdőcímét a 43–44-es című byte-okon tárolja. Ezek közül a 43-as – ha csak át nem írtuk – mindig 1-et tartalmaz. Mielőtt bármilyen belövésre váró programot írunk vagy töltünk a gépbe, ki kell adni egy: POKE 43,3 utasítást.

- A RUN parancs kiadása előtt olvassuk ki, és jegyezzuk fel a 45 és 46-os című byte-ok tartalmát.

?PEEK(45):?PEEK(46)

(Itt a program vége, illetve változók kezdete cím található.)

- Ha hibás működés miatt a RESET gombot meg kellett nyomnunk, három utasítás kiadásával újra elérhető lesz a program:

POKE 43,3

POKE 45, előzőleg kiolvasott érték

POKE 46, előzőleg kiolvasott érték

Ez az eljárás egyes fix címekre töltendő programok esetén nem használható, de ezeknél is meg lehet próbálni a következő módszert:

- A 45 és 46-os címekre továbbra is az előző módszernél írtak érvényesek.

- Továbbá a RUN parancs kiadása előtt ki kell olvasni és feljegyezni a felhasználó rendelkezésére álló memória (BASIC terület) első és második byte-jának tartalmát. (Ha nem tudjuk a címet, a 43–44-ről a következő utasítással kiolvashatjuk:

?PEEK(43)+256*PEEK(44) .)

- RESET után visszairjuk a feljegyzett értékeket:

POKE 45,... POKE 46,... valamint a BASIC terület első két előzőleg kiolvasott byte-ját is.

Befejezésül sok sikert kívánok mindenkinek az átalakításhoz, és kellemes újraindítását az így már el nem vesző programoknak. **Tamási András**, 2000 Szentendre, Lukács fivérek u. 21.

VC-20-as játékprogramokat szívesen cserélnék más programokra. A program neve: BLACK MAX, EARTH DEFENSE, STAR WARS, SIMON, KONG, FIRE, OTHELLO, RACE, SNAKE. Ezenkívül német nyelvű gépkönyvem teljébenlanám angol nyelvűért cserébe.

CSONKA ISTVÁN

Szolnok Pf. 5/K-I 5008

VZ 200 COLOR COMPUTER-t + 16 K-s bővítőt cserélnék SPECTRUM INTERFACE 1-re vagy MICRODRIVE-ra.

LITTER ZSOLT

Szigetvár, Sánc u. 3. 7900

VC 20-es Commodore számítógéphez való programért cserébe adok VC 64-es Commodore-programot. A program magnószalagon van.

MOLNÁR GÁBOR

Szeged, Retek u. 9/B. VI. em. 20.

TI-99/4A programokért bármilyen ismertebb személyi számítógéphez való programot (fénymásolaton vagy kazettán) cserélek minden szom-

PROGRAM CSERE-BERE

bat délelőtt a TI-99/4A személyi számítógép Klub-ban (1181 Batthyány u. 78/A.) és egyébként levélben.

KUN LÁSZLÓ

villamosmérnök tanár, klub-vezető
1205 Koppány u. 11.

Spectrumosok, figyelme! Két programot keresek: az egyik egy olyan compiler program, amely a Spectrum BASIC-jének minden utasítását ismeri, a másik a VU-CALC Cserébe összkomfortos másolóprogramot, szakmai és játékprogramot tudok küldeni.

MIZSEI BÉLA

villamosmérnök tanár,
Jászberény, Szabadság u. 11. 5100

A szerkesztő azért van,

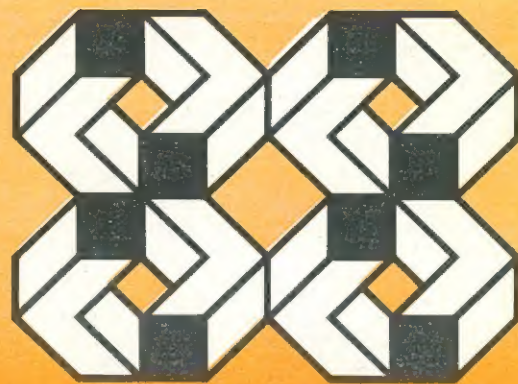
hogy a lap olyan legyen,

amilyenek az olvasói!

M08X

PROCAD-08

tervezői munkahely programcsomag



Az SZKI M08X személyi számítógépével és megfelelő, hazai gyártású perifériális berendezésekkel olyan olcsó tervezői munkahelyet lehet kialakítani, amely hatékonyan használható nyomtatott áramköri lapok tervezéséhez.

A tervezői munkahely funkciót a legelőnyösebben az SZKI nagyszámítógépes KENTAUR tervezőrendszeréhez kapcsolódva lehet kihasználni, azonban egyszerűbb tervezési feladatok önállóan is megoldhatók.

A PROCAD-08 programcsomag az M08X alapú tervezői munkahelyen a nyomtatott lapok tervezéséhez szükséges feladatokat végzi el.

FELHASZNÁLÁSI LEHETŐSÉGEK

A PROCAD-08 programcsomagot két jelretegű, furatgalvanizált, finomrajzolatú (1,27 mm-es raszteren huzalozott) nyomtatott lapok tervezéséhez lehet felhasználni. Segítségével a következő feladatok végezhetők el:

- elektromos (vagy logikai) kapcsolási rajz adatainak bevétele és a bekötöttség ellenőrzése,
- kézzel tervezett huzalozási rajz adatainak bevétele digitalizálóról,
- a KENTAUR rendszerben tervezett és módosításra szoruló nyomtatási kép adatainak a beolvasása,
- a nyomtatási kép megjelenítése képernyőn, grafikus üzemmódban, kiválasztási lehetőségekkel és grafikus funkciókkal
- a nyomtatási kép módosítása képernyőn.

FELHASZNÁLÁSI MÓDOK

1. A KENTAUR tervezőrendszerhez kapcsolódóan

- on-line, terminál üzemből, adatátviteli vonalon keresztül, az SZKI központi nagyszámítógépéhez kapcsolódóan,
- off-line, floppy diszk adathordozó közvetítéssel.

A KENTAUR rendszerben tetszőleges méretű és igen bonyolult tervezési feladatok is megoldhatók, de az adatok bevétele és

ellenőrzése, valamint a huzalozási kép módosítása sok kézi beavatkozást és sok gépidőt igényel.

A PROCAD-08 programok segítségével számos, emberi beavatkozást igénylő feladat a személyi számítógépen végezhető el, párbeszéd (interaktív), grafikus üzemmódban.

A tervezői munkahelyet bárhol az országban, közvetlenül a fejlesztői környezetben lehet telepíteni, ami a feladatok megoldását gyorsabbá és kényelmesebbé teszi és egyben a tervezői beavatkozások is könnyen és felelősségteljesen végezhetők el.

A PROCAD-08 és a KENTAUR programrendszerek, ill. az M08X és a központi nagyszámítógépek közötti feladatmegosztást vázlatosan az 1. ábra mutatja. (Az ábrán a KENTAUR rendszernek csak a PROCAD-08-cal kapcsolatos funkciói vannak feltüntetve.)

2. Önálló alkalmazásban

Ekkor a programokat a kézi tervezési folyamat kiegészítésére, ellenőrzési célokra, az adatok archiválására lehet felhasználni.

Ilyen módon egy célszerű alkalmazási folyamat:

- a) a kapcsolási rajz adatainak bevétele és ellenőrzése,
 - b) kézi nyomtatástervezés és bevitel digitalizálóról,
 - c) a kapcsolási rajz és a digitalizált huzalterv egybevetése,
 - d) a huzalterv módosítása, javítása képernyőn,
 - e) dokumentációkészítés: darabjegyzék, kötési lista.
- Ezután következik az adathordozók előállítása a rajzgép vezérléséhez (mágnesszalag) és a fúrógéphez (lyukszalag). Ezt a munkát azonban megfelelő periféria hiányában mindenképpen másutt kell elvégezni. (Az SZKI-ban közvetlen lehetőség van floppy diszkről Ferranti fotoplotterhez és Schmoll fúrógéphez a vezérlő szalagok előállítására.)

A PROCAD-08 ÖSSZETÉTELE

- ADEL programrendszer
- GRID programrendszer

ADEL programrendszer

Feladata a nyomvonaltervezéshez szükséges kapcsolási adatok előkészítése (bevitel, ellenőrzés, módosítás, adatbázis létrehozása, listázás).

Az egyes feladatokat különálló programok végzik, amelyek két adatbázison keresztül vannak egymással kapcsolatban (tervezési adatok és alkatrészkatalógus).

A programok párbeszéd (interaktív) üzemmódban kéri és dolgozza fel az adatokat. A formai hibák kijelzése bevitelkor megtörténik és így a javítás azonnal elvégezhető. Az adatok megadását a képernyőn látható mezőkijelölés segíti.

A rendszer főbb programjai és azok funkciói

•KARBE

- a kapcsolási rajz adatainak beolvasása,
 - az adatok ellenőrzése (szintaktikai és értéktartomány szerint),
 - a tervezési adatbázis megnyitása és kezdeti feltöltése.
- A 2. ábrán látható a képernyőkép a program indításáról és egy rövid részlet a párbeszéd adatbeviteléből.

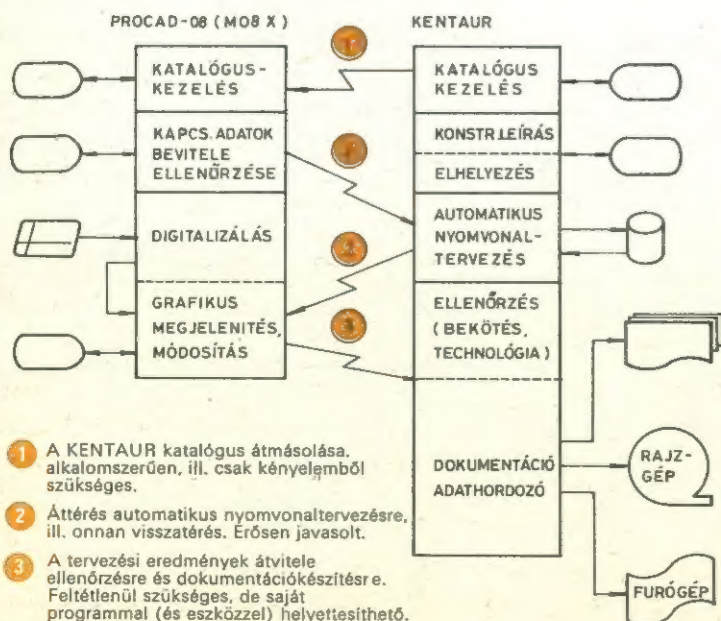
•KAREL

- szemantikai ellenőrzés az alkatrészkatalógus adatainak a felhasználásával,
- hibalista összeállítása, ill. a hibátlanág kinyilvánítása.

•KARMOD

- a szükséges módosítások elvégzése.

1. ábra



Sci-L

1011 Budapest

Iskola utca 10.

Telefonszám: 260-000

Telexszám: 22-4590

```

2. ábra

→ KARBE
→ DELTA
  ** OUTPUT LEMEZEJEGYSEG? **
  A vagy B
→ A
  ** UJ FELADAT? (I/N) **
→ I
  ** KONSTRUKCIO? **
  AAAAA
  TE2
  ** FELELOS,DATUM? **
  FFFFFFFFFFFFFFFFFFFFFF,XXXX.XX.XX
→ FELELOS_PETER,1983.11.22
  ** ELEMJEGYZEK BEVITELE **
  EEE[-EEE],A,TIPUS,T
  ** TIP=? **
→ 1-4,K,SN7415IAN
  ** TIP=? **
→ 5,K,SN7425N,T
  ** TIP=? **
→ 6,K,SN74820N
  ** TIP=? **
→ SAT08BI
  ** TUL HOSSZU TIPUSNEV **
  ** TIP=? **

```

```

→ KARBE
→ DELTA
  ** OUTPUT LEMEZEGYSEG? **
  A vagy B
→ A
  ** UJ FELADAT? (I/N) **
→ I
  ** KONSTRUKCIO? **
  AAAA
  TELZ
  ** FELELOS,DATUM? **
  FFFFFFFFFFFFFFFFFFFFFF,XXXX,XX,XX
  FELELOS.PETER,1983.11.22
  ** ELEMJEGYZEK BEVITELE **
  EEEI-EEI,A,TIPUS,T
  ** TIP=? **
→ 1-4,K,SN74151AN
  ** TIP=? **
→ 5,K,SN7425N,T
  ** TIP=? **
→ 6,K,SN74520N
  ** TIP=? **
→ SATOBBI
  NN TUL HOSSZU TIPUSNEV NN
  ** TIP=? **

```

3. abra

SN74S30

U SN74S30 L
K SN74S30N
F 14 00 1 9
L 11 12 3 4 5 6 11 12 8
E 1:BE BE BE BE BE BE 0V:0V KI NC NC BE BE NC TA:+5V

SN74S04

U SN74S04 L
K SN74S04N
F 14 00 6 2
L 11 12 3 4 5 6
E 1:BE KI BE KI BE KI 0V:0V KI BE KI BE KI BE TA:+5V

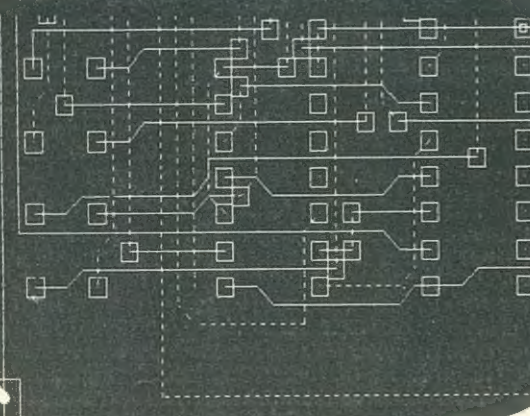
SN74S00

U SN74S00 L
K SN74S00N
F 14 00 4 3
L 11 12 3 4 5 6
E 1:BE BE KI BE BE KI 0V:0V KI BE BE KI BE BE TA:+5V

```
SN74S04
C SN74S04 L
N SN74S04N
F 1 4 0 9
E 1 2 3 4 5 6 11 12 8
I BE BE BE BE BE BE 0V:0V KI NC NC BE BE NC TA:+SV

SN74S04
C SN74S04 L
N SN74S04N
F 1 4 0 2
E 1 2 3 4 5 6 11 12 8
I BE KI BE KI BE KI 0V:0V KI BE KI BE KI TA:+SV

SN74S00
C SN74S00 L
N SN74S00N
F 1 4 0 3
E 1 2 3 4 5 6 11 12 8
I 0V:0V KI BE BE KI BE BE KI BE BE TA:+SV
```



- a bevitt tervezési adatok rendezett kiírása (elemjegyzék, csatlakozókiosztás, kötési lista),
- az alkatrészkatalógus adatainak kiírása (tartalomjegyzék, részletes adatok).

PKATAL

GRID programrendszer

A nyomtatott lap huzalozási képe kétféleképpen készülhet el és juthat a M08X személyi számítógép adattárába:

■ automatikusan, számítógéppel, a KENTAUR rendszerben, ahonnan az eredmény adatátviteli vonalon keresztül hozzátartozó át az M08X számítógépbe.

A huzalozási kép megjelölése a személyi számítógép képernyőjén történik, az ún. grafikus üzemmódban. A képernyő ilyenkor 512×512 képpontra bontható és a pontokkal a huzalozást vonalas ábra alakjában meg lehet jeleníteni.

A huzalozási rajz szükséges módosítását és kiegészítését a képernyőn, a billentyűk és a mozgatható kurzor segítségével lehet elvégezni.

A véglegesnek tartott képet vissza lehet küldeni a központi gépre, további ellenőrzés, majd felhasználás céljából.

• DIGI

- a grafikus file feltöltése digitalizálóról beolvasott adatokkal.

- a grafikus file megjelenítése képernyőn, azonosítók szerinti szelekcióval vagy anélkül (azonosítók: rétegszám, jelsorszám, furatkód stb.),

- **grafikus funkciók:** ablakozás, eltolás, nagyítás-kicsinyítés, kurzor-mozgatás, kurzor-lekérdezés, szaggatott vonalképezés stb.

- rajzelemek (vonalak, furatok) leírása és törlése, kurzorpozíció és azonosító szerint.

A 4. ábrán egy nyomtatott lap huzalozási képe látható, ahol az egyik réteg vezetőkei szaggatottan vannak ábrázolva.

A PROCAD-08 programcsomag az SZKI személyi számítógépcsaládjának M08X gépén használható.

A szükséges hardver konfiguráció

■ M08X központi egység (64 KByte, képernyő, billentyűzet)

■ **soroscsatolás-kártya (SIO)**

■ **párhuzamos csatolás-kártya (PIO)**

■ grafikus kártya

■ mátrixnyomtató (MP-80)

- 2 x 8" hajlékony mágneslemez-es egység (MF-6400, MOM)

■ digitalizáló (RA-06, Fok Gyem)

■ opcióként: rajzgép (OH-860, RADELKISZ)

2 db MODEM (AM-12TD, Orion).

A szükséges szoftver

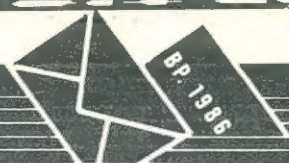
■ PROPOS-8 V.1.0 operációs rendszer (ADEL-hez)

■ **GRAFOS-8** grafikus operációs rendszer (GRID-hez)

- GR80 grafikus handler

- opcióként: TD-8160 szinkron terminál-emulátor.

A programcsomag PASCAL nyelven készült, assembly nyelvű betétekkel.



Igérletünkhöz híven márciusban ismét megtartjuk telefonos szervizszolgálatunkat.

MÁRCIUS 4-ÉN DÉLUTÁN 14-18 ÓRA KÖZÖTT HÍVHATNAK BENNÜNKET OLVASÓINK KÉRDÉSEIKKEL.

A TELEFONSZÁMOK ÉS A GÉPTÍPUSOK FELOSZTÁSA A KÖVETKEZŐ:

SPECTRUM-TÉMÁBAN A 403-743-AT!

COMMODORE-ÜGYBEN A 403-755-ÖT!

PRIMÓVAL KAPCSOLATBAN A 403-797-ET!

MÁS TÉMAJÚ KÖZLENDŐJÜKKEL A 403-744-ET HÍVJÁK!

BIT-LET TELEFONSZERVIZ

MÁRCIUS 4-ÉN DÉLUTÁN!

HÍVJA ÖN IS!

Kedves BIT-LET!

Kérdésem az, hogy most lehetőségem nyílt, hogy egy számítógépet hozassak, de nem tudom eldönteni, hogy melyet! A Spectrum és a Commodore 64 jöhet szóba. Eddig csak ZX81-en dolgoztam és a Z80 programozásához is értek egy keveset. Ezért és az ára miatt is a Spectrumot választanám. A későbbi évekre gondolva (talán a számítástechnikával fogok foglalkozni) viszont jobb lenne a C 64.

Perity Antal, Budapest XI., Szakasits Á. út 32/C

Ilyen és hasonló kérdésekre határozott választ nem adunk. A lap többek között azért is van, hogy az abból szerzett információk segítségével ezt ki-kí a maga igényei, lehetőségei szerint döntse el.

A tavaly decemberi BIT-LET-ben jelent meg a ZX81 szintetizátorprogram. Ennek komoly szépséghibája, hogy a magasabb hangok hamisak. A szintetizátor „felhangolásához” a II. táblázatot kell módosítanunk. Az ebben található „hangfrekvenciák”, azaz osztásarányok megfelelnek ugyan az elméletileg kiszámolhatóknak, de belső késleltetések miatt ezeket módosítani kell. Tapasztalati úton a „hangmagasság” oszlop a következőképp alakul:

255-170 között nincs változás

161-107-ig eggyel kevesebb (kivéve 114 helyett 112)

101-80-ig kettővel kevesebb

a többi hárommal kevesebb az eredetinel. Az utolsó, a 26-os kódú billentyűnél is 61 a hangmagasság!

A módosított adatokat célszerűen az eredeti programmal lehet beírni, majd rögtön rögzíteni kazettán.

Sok sikert a kipróbáláshoz!

Keresztény István, 2120 Dunakeszi, Barátság u. 21.

Nem próbáltuk ki a módosítást, de közzétesszük azzal a megjegyzéssel, hogy a gépek szubjektív tulajdonságai is okozhatnak minimális eltérést.

Tisztelt Szerkesztőség!

S.O.S.

SEGÍTSÉG!

HELP!

Májusban vettem - újonnan - Münchenben egy ZX81-et egy 32 K-s memóriával, de valami nem stimmel.

10 INPUT X

20 LET Y=X-0.5

30 FOR F=X TO 0 STEP -0.1

40 SCROLL

50 PRINT F, (INT((F-0.1)*10))/10

60 IF F=Y THEN GOTO 105

70 IF INKEY\$="U" THEN GOTO 10

80 IF INKEY\$="S" THEN GOTO 120

90 NEXT F

100 GOTO 10

105 SCROLL

110 PRINT "IF"

120 LET Y=F-0.5

130 PAUSE 4E4

140 GOTO 70

"U" gomb lenyomására új Y értéket kér, "S" gombra megáll és bármely gombra folytatja (kivéve U, ill. S gombra).

A probléma az, hogy X értékétől függően más-más számsorozatot ír ki. A 60-as sor is csak elvétve hajtódik végre, pl. ha X=2 vagy 8, vagy 32, de ekkor is csak egyszer.

A következő táblázatban leírtam azokat az értékeket, amelyeket az 50-es sorban a PRINT második részében ír ki:

X=2	9	10	11	12	14	15	30	31
1.9	8.9	9.9	10.9	11.9	13.9	14.9	29.9	30.9
.	8.7	9.7	10.7	11.7	13.8	14.8	29.8	30.8
.	8.6	9.6	10.6	11.6	13.6	14.6	29.7	30.7
1.3							29.6	30.6
1.1							29.3	30.3
1.0								
0.9								
0.8								
0.7								
0.7								
0.6								

Ha x=3-8 között van akkor jó.

Az 50-es sorra azért volt szükség, mert a kezdőértéktől függően egy adott F értéktől 8 számjegy pontossággal (pontoslatlanul) írja ki, de abban a programban, amelyben alkalmazni akartam, csak egytizedes pontosságra volt szükségem. Hogy miért vált át nyolc számjegyes kijelzésre, azt sem értem, sejtem, hogy ez a három dolog összefügg. Az alapgép is és memóriával is így viselkedik, akkor is, ha nem POKE-olom 32 K-re. PRINT PEEK 54-re 136-ot ad, tehát új ROM-os. Ha megfordítom a számlálás irányát vagy kivonás helyett összeadást írok, akkor is hasonlóan számol.

Mi a hiba? A gép? A ROM? A Z80? Kíváncsi idegességgel várom a választ!

Katona László, 1033 Folyamőr u. 2.

A hiba okát önmagában kell keresnie. A küldött program elemzésével nagyon közel járt az igazsághoz, de éppen azt nem meri feltételezni, hogy a gép SZÁMABRÁZOLÁSA és MŰVELETEINEK ELVÉGZÉSE nem pontos! Márpedig azt már elemi szinten is tudni illik, hogy egy nem egész típusú változó értéke igen valószínűtlen, hogy pontos. Ennek az az oka, hogy a gép véges darab biten képes tárolni a számokat, ez pedig nem mindig egzakt érték. (Gondoljon arra, hogy π értékét még senki sem látta pontosan leírva!) A 0.1 pedig éppen nem véges a kettes számrendszerben.

Hasonló okokból a számítások során is halmozódhat egy bizonyos hiba, ezek együttes hatása természetesen előre egyszerűen nem határozható meg.

Igy azután két, nem egész típusú számról sohasem várhatjuk el, hogy egyenlők legyenek, ha elméletileg egyezniük kellene. Ez az egyik oka, hogy egyes gépeken létezik egész típus.

KERAVILL MEV
ELEKTRONIKAI
MÁRKABOLT 
BP.V. MŰZEUM krt. 11.

MIKROELEKTRONIKA:
A JÖVŐ A JELENBEN.

FÉLVEZETŐK,
INTEGRÁLT ÁRAMKÖRÖK,
MIKROPROCESSZOROK
ÉS CSATLAKOZÓK.
SZAKTANÁCSADÁS, CSOMAGKÜLDŐ SZOLGÁLAT.

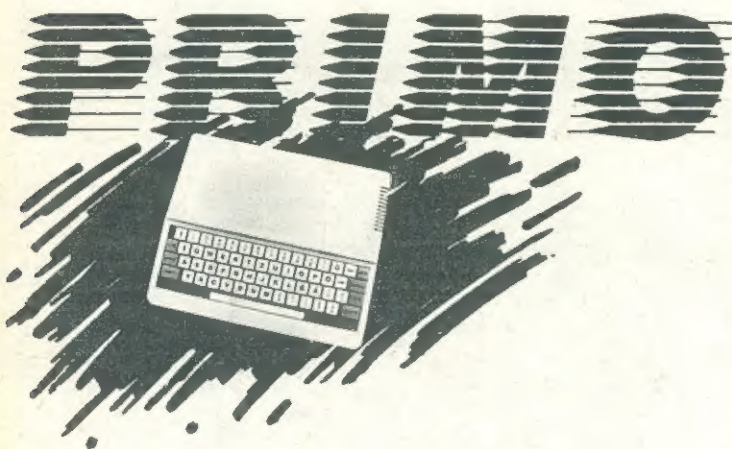
HT-GÉPNYERŐ

Bár már a 3. feladat és a „nagy” feladat megoldásai is megérkeztek, most mégis a második feladatra küldött megoldásokról szólnak néhány értékelő szót:

A feladat nehézségnek bizonyult. Sok szakkör helytelenül értelmezte a feladat 2. kívánságát, s még többen voltak, akik az írandó program célját nem tartották eléggé szem előtt. A főbb értékelési szempontok (fontossági sorrendben):

1. a program működőképessége
2. a lift működésének jó megfigyelhetősége és kipróbálhatósága
3. a lift működési algoritmusának hibátlansága és „jósa”
4. valóságűség
5. formai kivitelezés

Gyakori hibák: sok csapat küldött be hibásan működő programot, vagy olyat, amelyben a lift algoritmus egyértelműen hibás. Több olyan program is érkezett, amely a megfigyelhetőség és kipróbálhatóság alacsony színvonalára miatt használhatatlan. Elég általános hiba, hogy a csapatok nagy része sokkal nagyobb figyelmet szentelt a formai kivitelezésnek, mint a 2. és 3. szempontnak, így pl. a két legprofibb kiviteli grafikával rendelkező programban lényeges egyéb hiányosságok voltak. Az is jellemző volt, hogy általában még a működésképtelen, használhatatlan programok is zenéltek. Jó lenne ha az ilyen programokat beküldő szakkörök megjegyeznék, hogy egy programnak sokkal fontosabb a tudása, működése, mint a megjelenítés és a hang, amiket általában csak a többi rész készenléte és kipróbálása után érdemes a programba tenni. Ha pedig a munkamegosztás során ezek a részek külön ember(ek)hez kerülnek (ami helyes is), akkor ne a legjobb programozóknak adjuk ki ezt a részt! Persze, mindezek nemcsak erre a pályázatra vonatkoznak, hanem bármilyen más munkára. S még azt is megjegyezzük, hogy azért bizonyos minőségű megjelenítés ennél a feladatnál – a 2. szempont miatt – szükséges volt. Végül is a beérkezett 47 programból 12 volt jónak mondható, ezeket a szakköröket most nem soroljuk fel.



NYERŐ NYERŐ NYERŐ

1. FELADAT MEGOLDÁSA:

Kezdő Ödön tud nyerni, mégpedig elsőnek a 6-ot kell mondania.

Nézzük, miket válaszolhat Második Odik:

1. Ha 36-ot vagy 24-et mond, akkor utána Ödön tudja a 2 közül a másikat mondani, s ezután Odik kénytelen kimondani a 72-öt, s így veszít.

2. Ha Odik 9-et mond, erre Ödön 8-at válaszol, ha pedig Odik mond 8-at, akkor Ödön mondja a 9-et. E lépéspár után a megmaradó számok,

amiket még mondhatnak: 12, 18, 24, 36 és 72. Odik következik. Ha 72-öt mond, veszít, ha 24 és 36 valamelyikét mondja, az 1. pont miatt szintén veszít. Ha a 12 és 18 valamelyikét mondja, akkor Ödön nyugodtan rávághatja a 2 közül a másikat, s így most már Odik csak a 24, 36 és 72 közül választhat, tehát veszít.

3. Odik a 18-at mondja. Ödön válasza: 4. Ha Odik mond 4-et, Ödön 18-at válaszol. A lehetséges számok ezek után: 8, 12, 24, 36, 72. Ha Odik a 24, 36, 72 számok valamelyikét mondja, akkor veszít (lásd 1. pont!), ha a 8 és 12 valamelyikét, akkor Ödön a kettő közül a másikat válaszolja rá, s így már Odiknak mindenképpen a 24, 36, 72 közül kell valamit mondani, tehát veszít.

4. Még egy lehetőséget nem tárgyaltunk, ha Odik a 6-ra 12-öt válaszol. Erre Ödön 9-et mond. A lehetséges számok: 8, 18, 24, 36, 72. Ha Odik a 24, 36, 72 valamelyikét mondja, akkor veszít. Ha a 8, 18 valamelyikét mondja, akkor Ödön a 2 közül a másikat válaszolja, s így Odiknak megintcsak a 3 „rossz” szám marad.

Láttuk tehát, hogy Odik bármit is válaszol, Ödön a fenti stratégiát követve mindenképpen nyerni tud. Ezzel állításunkat beláttuk.

2. FELADAT:

Játsszuk tovább az osztójátékot! Állításunk a következő: bármilyen 1-nél nagyobb számot választanak alapszámmal a játékosok, mindenképpen a kezdőnek van nyerő stratégiája. Olvasóink feladata ennek az állításnak a (logikai úton történő) bebizonyítása!

2. EZ A MENŐ

PRIMO NYERŐ

Kérjük levégni és a levélre felragasztani!
Beküldési határidő: március 12.